Optimal navigation of autonomous systems through dynamic environments

Tim Mercy* Goele Pipeleers*

Safe operation of autonomous systems demands a collision-free motion trajectory at every time instant. This abstract presents a method to generate time-optimal motion trajectories for autonomous systems moving through an environment with both stationary and moving obstacles.

The motion trajectory is obtained by solving an optimal control problem, which is translated into an optimization problem. Each trajectory will connect the current state of the autonomous system with the desired end state, which is imposed by the initial and final constraints. The autonomous system itself is represented by a kinematic vehicle model, which determines constraints on the velocity, acceleration,... Furthermore, there are anti-collision constraints which express that the shape of the autonomous system must not overlap with the shape of any obstacle. The proposed approach formulates these constraints by using the separating hyperplane theorem [1], which states that two non-intersecting convex sets can always be separated by a hyperplane. These constraints also contain a safety factor, which ensures that the system keeps a safe distance from all obstacles in the environment. Finally, the goal of the optimization problem is to minimize the motion time of the system.

In general, autonomous systems move through dynamic and uncertain environments, which complicates the basic optimization problem. First of all, the basic problem is extended with a motion model to predict obstacle movements, using the estimates of their current position and velocity. Since the environment is uncertain, reality will differ from the predictions. Furthermore, new obstacles can show up. Therefore, the optimal control problem is solved with a receding horizon, to always take the latest information into account. As a consequence, the approach must be suitable for real-time optimization. To obtain an optimization problem which is small-scale and easy to solve, the method (i) uses a spline parameterization of the motion trajectory; and (ii) exploits spline properties to reduce the number of constraints.

The proposed approach is implemented as part of a general spline-based motion planning toolbox[1]. This toolbox contains examples demonstrating which types of problems the method can handle and showing different vehicle models, including a holonomic model and nonholonomic ones like a Dubins car, an AGV with rear-wheel steering, a quadrotor,...

To experimentally validate the method, this toolbox is linked to an experimental set-up containing a *KUKA youBot*. The developed method computes a motion trajectory. The derivative of this trajectory gives velocity setpoints which are send to the *youBot* in feedforward. Because of the small solving time additional position control is not necessary, if the robot deviates from its desired trajectory, the method takes this into account in the next time step. During the experiments the platform moved through an environment containing three obstacles. The average solving time of the optimization problem was 0.05s, which is sufficiently fast for correcting deviations from the desired trajectory.

References

[1] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge, UK: Cambridge University Press, 2004.

*Department of Mechanical Engineering, KU Leuven, `<firstname>.<surname>@kuleuven.be`

[1] `https://github.com/meco-group/omg-tools`